AppleWorks Forum

A Publication of **NAUG:** ***The National AppleWorks Users Group***

Volume X, No. 3 **Five Dollars**

How to Preview Your Picture Fonts

by Bill Thompson and Jack Fowler

This article is part of a continuing series that describes projects you can create with AppleWorks and TimeOut SuperFonts. This month you will learn how to use the AppleWorks spreadsheet module to create documents that let you preview your picture fonts.

AppleWorks users enjoy the challenge of discovering new and productive ways to use their computers. When you put your mind to it, you can do almost anything with AppleWorks.

TimeOut enhancements play an important role in extending this functionality. We particularly like SuperFonts, which lets us create attractive and unusual documents. Part of SuperFonts' usefulness comes from the availability of picture fonts like Mobile and Cairo that make it easy to add graphics to your documents. *[Ed: Each character in a "picture font" is a graphic. An example of a picture font appears in Figure 1.]*

Unfortunately, picture fonts pose three problems. First, it is difficult to find the picture you need. This is more of a problem than most users realize because many fonts include three sets of "characters" (either letters or graphics). The first set is the "standard" character set; the two alternate sets are the "extended" character sets. *[Ed: See the sidebar entitled "Alternate Character Sets" for a description of these sets and how to use them in SuperFonts.]*

Second, there is no easy way to preview the size and shape of each character.

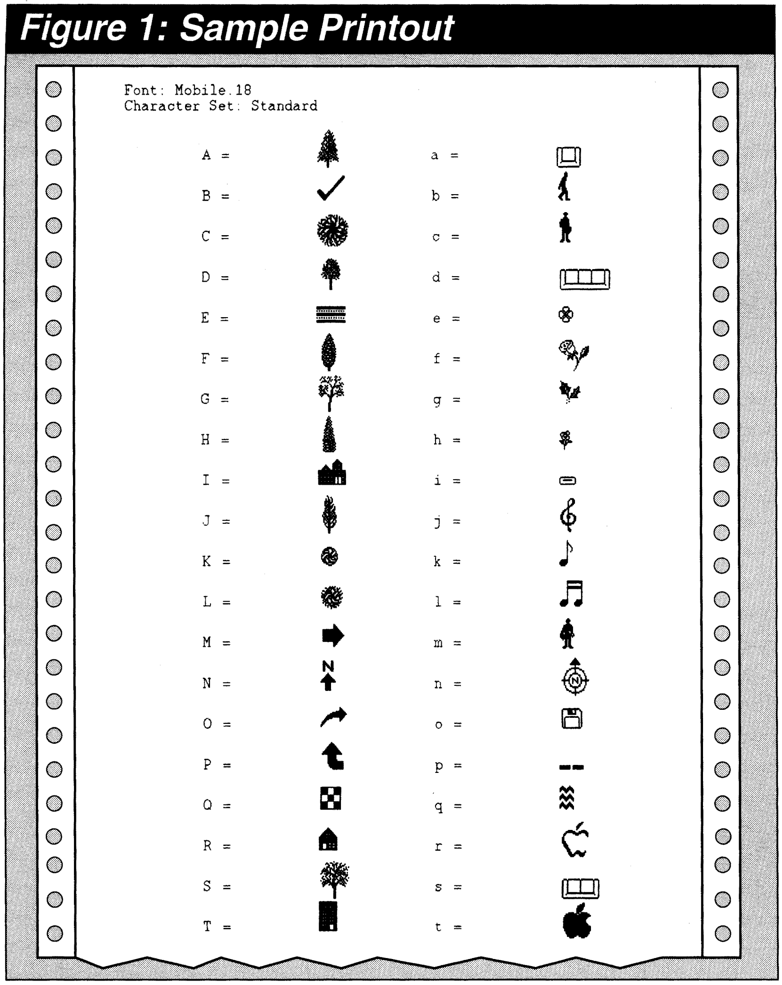
Third, there is no direct correspondence between any key on your Apple keyboard and the character that prints in your document. Unlike letter fonts, pressing the "A" key in Mobile does not generate an "A" but a picture of an evergreen tree.

By the time you finish this article you will be able to print samples of all the characters in any picture font. Your output will look like the example in *Figure 1.*

The procedure we outline uses two AppleWorks templates. One is

Table of Contents

Figure 1: Sample Printout



Font: Mobile.18
Character Set: Standard

```
A =        a =
B =   ✓    b =
C =        c =
D =        d =
E =        e =
F =        f =
G =        g =
H =        h =
I =        i =
J =        j =
K =        k =
L =        l =
M =        m =
N =        n =
O =        o =
P =        p =
Q =        q =
R =        r =
S =        s =
T =        t =
```

the KEY.REF.MAKER spreadsheet template in *Figure 2*. You will use that spreadsheet to prepare the KEY.REF word processor template in *Figure 3* that prints your final output.

When you are done, you can use the templates to print all the characters in any SuperFonts-compatible font on your system.

The KEY.REF.MAKER Template: Section A

You will begin by creating the KEY.REF.MAKER spreadsheet template in *Figure 2*. The template has three sections. Section A includes cells A1 through W27; Section B, cells A31 through W56; Section C, cells A61 through W86.

Each section, which includes SuperFonts commands and keyboard characters, corresponds to one of the three character sets available for a font. Later you will copy each area of the template to a word processor file and use SuperFonts to print the characters available in that set.

Follow these steps to begin the spreadsheet:

1. Launch AppleWorks and create a new spreadsheet named "KEY.REF.MAKER".
2. Use Apple-L to narrow the spreadsheet's columns to the column widths in *Figure 4*. *[Ed: You reduce all the columns to seven characters and then narrow the selected columns further.]*
3. Use Apple-V to set "Recalculate" "Frequency" to "Manual".
4. Type <1> in cell A2. *[Ed: You must press Shift-" before entering many of the labels in this template.]*
5. Use Apple-C to copy cell A2 "Within worksheet" to cells A3 through A27. This fills the range of cells with "<1>". Later, when you copy the spreadsheet block to the word processor document, SuperFonts will interpret the label as a Change Font Command. *[Ed: Font <1> will be the Courier.12 font in your printout.]*
6. Copy cells A2 through A27 to the clipboard as a "Block". Then copy the block into cells G2, M2, and S2. You need four columns to accommodate all the keyboard keys.
7. Enter "<2>" in cell D2. *[Ed. This will switch to the picture font in the word processor document you will create later in this tutorial.]*
8. Copy cell D2 "Within worksheet" to cells D3 through D27.
9. Copy cells D2 through D27 to the clipboard as a "Block". Then copy the block to cells J2, P2, and V2.

That completes your entry of the Change Font Commands. Now you will enter the keyboard characters and copy them into Section A of the spreadsheet. Continue as follows:

1. Type the upper-case letters of the alphabet starting with "A" in cell B2, "B" in cell B3, and so on through cell B27.
2. Copy cells B2 through B27 "Within worksheet" into cells E2 through E27.

3. Enter the lower-case letters of the alphabet in cells H2 through H27.
4. Copy cells H2 through H27 “Within worksheet” into cells K2 through K27.
5. Enter the numbers “1” through “Ø” in cells N2 through N11. You can enter the numbers as values or labels.
6. In cells N12 through N21, use the Shift Key to enter the symbols that appear above the numbers in the number row on the keyboard. *[Ed: Press Shift-" before each entry.]*
7. In cells N22 through N27 and cells T2 through T17, enter the remaining symbols and punctuation marks on your keyboard. Remember to press the Shift Key to get the “upper case” symbols and marks. Also remember to enter the characters on the keys immediately to the left and right of the Space Bar.
8. Copy cells N2 through N27 “Within worksheet” into cells Q2 through Q27.
9. Copy cells T2 through T17 “Within worksheet” into cells W2 through W17.

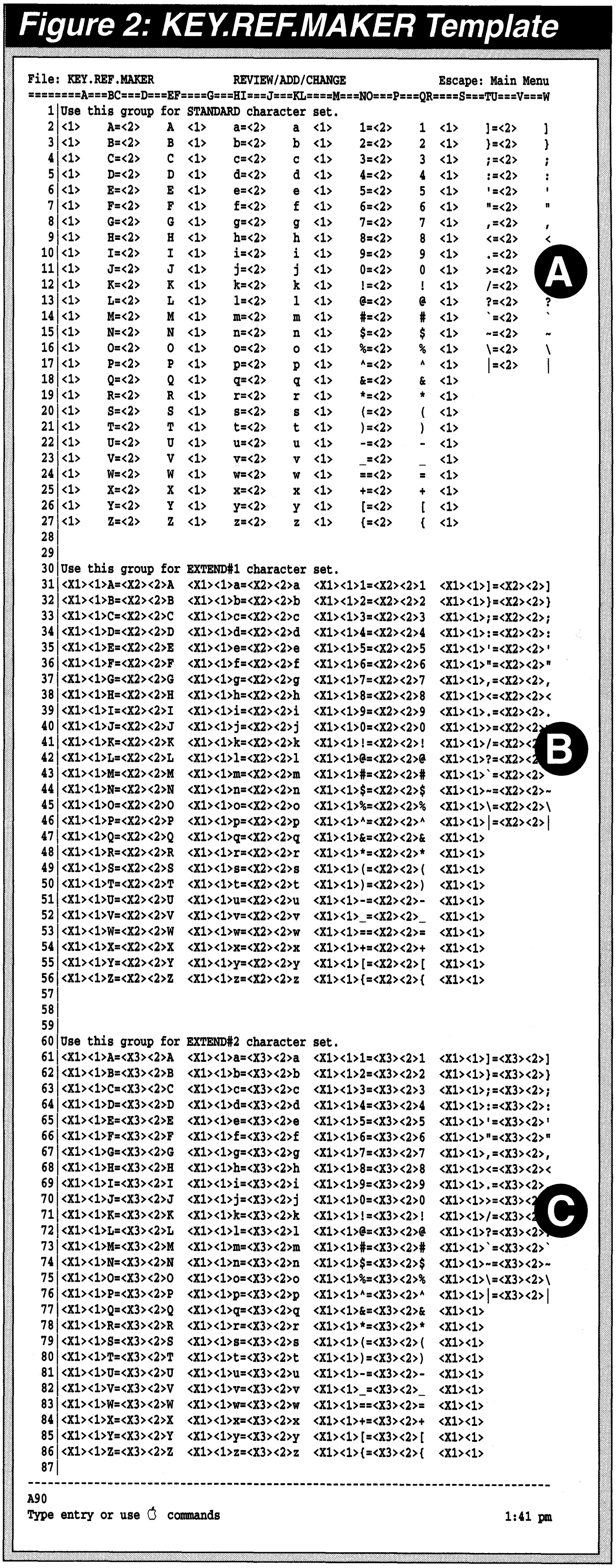
You have now entered all of the keyboard characters. Follow these steps to complete Section A:

1. Type “=” in cell C2.
2. Copy cells C2 “Within worksheet” to cells C3 through C27.
3. Copy cells C2 through C27 to the clipboard as a “Block”. Then copy the block to cells I2, O2, and U2.
4. Beginning in cell A1 type “Use this group for STANDARD character set.”
5. Save your file to disk. Your screen should look like the example in *Figure 5.*

Creating Sections B and C

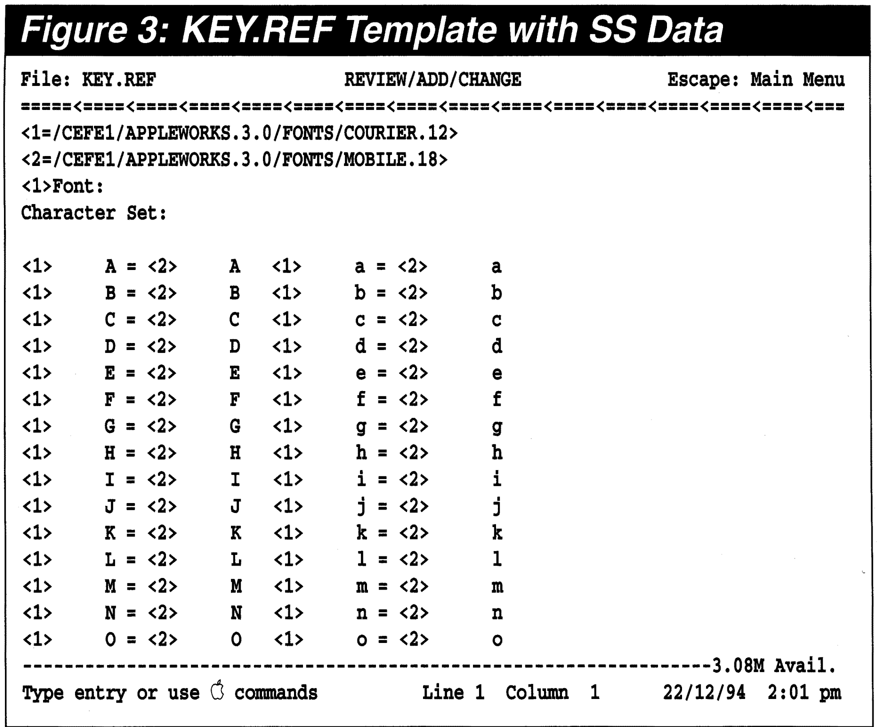
Now you will copy the cells from Section A into Sections B and C. Then you will modify the two new sections so they include the appropriate commands for the two extended character sets. Follow these steps:

Figure 2: KEY.REF.MAKER Template



```
File: KEY.REF.MAKER               REVIEW/ADD/CHANGE                Escape: Main Menu
========A===BC===D===EF====G===HI===J===KL====M===NO===P===QR====S===TU===V===W
  1|Use this group for STANDARD character set.
  2|<1>    A=<2>   A  <1>    a=<2>    a  <1>    1=<2>    1  <1>    ]=<2>    ]
  3|<1>    B=<2>   B  <1>    b=<2>    b  <1>    2=<2>    2  <1>    }=<2>    }
  4|<1>    C=<2>   C  <1>    c=<2>    c  <1>    3=<2>    3  <1>    ;=<2>    ;
  5|<1>    D=<2>   D  <1>    d=<2>    d  <1>    4=<2>    4  <1>    :=<2>    :
  6|<1>    E=<2>   E  <1>    e=<2>    e  <1>    5=<2>    5  <1>    '=<2>    '
  7|<1>    F=<2>   F  <1>    f=<2>    f  <1>    6=<2>    6  <1>    "=<2>    "
  8|<1>    G=<2>   G  <1>    g=<2>    g  <1>    7=<2>    7  <1>    ,=<2>    ,
  9|<1>    H=<2>   H  <1>    h=<2>    h  <1>    8=<2>    8  <1>    <=<2>    <
 10|<1>    I=<2>   I  <1>    i=<2>    i  <1>    9=<2>    9  <1>    .=<2>
 11|<1>    J=<2>   J  <1>    j=<2>    j  <1>    0=<2>    0  <1>    >=<2>
 12|<1>    K=<2>   K  <1>    k=<2>    k  <1>    !=<2>    !  <1>    /=<2>
 13|<1>    L=<2>   L  <1>    l=<2>    l  <1>    @=<2>    @  <1>    ?=<2>    ?
 14|<1>    M=<2>   M  <1>    m=<2>    m  <1>    #=<2>    #  <1>    `=<2>    `
 15|<1>    N=<2>   N  <1>    n=<2>    n  <1>    $=<2>    $  <1>    ~=<2>    ~
 16|<1>    O=<2>   O  <1>    o=<2>    o  <1>    %=<2>    %  <1>    \=<2>    \
 17|<1>    P=<2>   P  <1>    p=<2>    p  <1>    ^=<2>    ^  <1>    |=<2>    |
 18|<1>    Q=<2>   Q  <1>    q=<2>    q  <1>    &=<2>    &  <1>
 19|<1>    R=<2>   R  <1>    r=<2>    r  <1>    *=<2>    *  <1>
 20|<1>    S=<2>   S  <1>    s=<2>    s  <1>    (=<2>    (  <1>
 21|<1>    T=<2>   T  <1>    t=<2>    t  <1>    )=<2>    )  <1>
 22|<1>    U=<2>   U  <1>    u=<2>    u  <1>    -=<2>    -  <1>
 23|<1>    V=<2>   V  <1>    v=<2>    v  <1>    _=<2>    _  <1>
 24|<1>    W=<2>   W  <1>    w=<2>    w  <1>    ==<2>    =  <1>
 25|<1>    X=<2>   X  <1>    x=<2>    x  <1>    +=<2>    +  <1>
 26|<1>    Y=<2>   Y  <1>    y=<2>    y  <1>    [=<2>    [  <1>
 27|<1>    Z=<2>   Z  <1>    z=<2>    z  <1>    {=<2>    {  <1>
 28|
 29|
 30|Use this group for EXTEND#1 character set.
 31|<X1><1>A=<X2><2>A  <X1><1>a=<X2><2>a  <X1><1>1=<X2><2>1  <X1><1>]=<X2><2>]
 32|<X1><1>B=<X2><2>B  <X1><1>b=<X2><2>b  <X1><1>2=<X2><2>2  <X1><1>}=<X2><2>}
 33|<X1><1>C=<X2><2>C  <X1><1>c=<X2><2>c  <X1><1>3=<X2><2>3  <X1><1>;=<X2><2>;
 34|<X1><1>D=<X2><2>D  <X1><1>d=<X2><2>d  <X1><1>4=<X2><2>4  <X1><1>:=<X2><2>:
 35|<X1><1>E=<X2><2>E  <X1><1>e=<X2><2>e  <X1><1>5=<X2><2>5  <X1><1>'=<X2><2>'
 36|<X1><1>F=<X2><2>F  <X1><1>f=<X2><2>f  <X1><1>6=<X2><2>6  <X1><1>"=<X2><2>"
 37|<X1><1>G=<X2><2>G  <X1><1>g=<X2><2>g  <X1><1>7=<X2><2>7  <X1><1>,=<X2><2>,
 38|<X1><1>H=<X2><2>H  <X1><1>h=<X2><2>h  <X1><1>8=<X2><2>8  <X1><1><=<X2><2><
 39|<X1><1>I=<X2><2>I  <X1><1>i=<X2><2>i  <X1><1>9=<X2><2>9  <X1><1>.=<X2><2>.
 40|<X1><1>J=<X2><2>J  <X1><1>j=<X2><2>j  <X1><1>0=<X2><2>0  <X1><1>>=<X2><2
 41|<X1><1>K=<X2><2>K  <X1><1>k=<X2><2>k  <X1><1>!=<X2><2>!  <X1><1>/=<X2><2
 42|<X1><1>L=<X2><2>L  <X1><1>l=<X2><2>l  <X1><1>@=<X2><2>@  <X1><1>?=<X2><2
 43|<X1><1>M=<X2><2>M  <X1><1>m=<X2><2>m  <X1><1>#=<X2><2>#  <X1><1>`=<X2><2>
 44|<X1><1>N=<X2><2>N  <X1><1>n=<X2><2>n  <X1><1>$=<X2><2>$  <X1><1>~=<X2><2>~
 45|<X1><1>O=<X2><2>O  <X1><1>o=<X2><2>o  <X1><1>%=<X2><2>%  <X1><1>\=<X2><2>\
 46|<X1><1>P=<X2><2>P  <X1><1>p=<X2><2>p  <X1><1>^=<X2><2>^  <X1><1>|=<X2><2>|
 47|<X1><1>Q=<X2><2>Q  <X1><1>q=<X2><2>q  <X1><1>&=<X2><2>&  <X1><1>
 48|<X1><1>R=<X2><2>R  <X1><1>r=<X2><2>r  <X1><1>*=<X2><2>*  <X1><1>
 49|<X1><1>S=<X2><2>S  <X1><1>s=<X2><2>s  <X1><1>(=<X2><2>(  <X1><1>
 50|<X1><1>T=<X2><2>T  <X1><1>t=<X2><2>t  <X1><1>)=<X2><2>)  <X1><1>
 51|<X1><1>U=<X2><2>U  <X1><1>u=<X2><2>u  <X1><1>-=<X2><2>-  <X1><1>
 52|<X1><1>V=<X2><2>V  <X1><1>v=<X2><2>v  <X1><1>_=<X2><2>_  <X1><1>
 53|<X1><1>W=<X2><2>W  <X1><1>w=<X2><2>w  <X1><1>==<X2><2>=  <X1><1>
 54|<X1><1>X=<X2><2>X  <X1><1>x=<X2><2>x  <X1><1>+=<X2><2>+  <X1><1>
 55|<X1><1>Y=<X2><2>Y  <X1><1>y=<X2><2>y  <X1><1>[=<X2><2>[  <X1><1>
 56|<X1><1>Z=<X2><2>Z  <X1><1>z=<X2><2>z  <X1><1>{=<X2><2>{  <X1><1>
 57|
 58|
 59|
 60|Use this group for EXTEND#2 character set.
 61|<X1><1>A=<X3><2>A  <X1><1>a=<X3><2>a  <X1><1>1=<X3><2>1  <X1><1>]=<X3><2>]
 62|<X1><1>B=<X3><2>B  <X1><1>b=<X3><2>b  <X1><1>2=<X3><2>2  <X1><1>}=<X3><2>}
 63|<X1><1>C=<X3><2>C  <X1><1>c=<X3><2>c  <X1><1>3=<X3><2>3  <X1><1>;=<X3><2>;
 64|<X1><1>D=<X3><2>D  <X1><1>d=<X3><2>d  <X1><1>4=<X3><2>4  <X1><1>:=<X3><2>:
 65|<X1><1>E=<X3><2>E  <X1><1>e=<X3><2>e  <X1><1>5=<X3><2>5  <X1><1>'=<X3><2>'
 66|<X1><1>F=<X3><2>F  <X1><1>f=<X3><2>f  <X1><1>6=<X3><2>6  <X1><1>"=<X3><2>"
 67|<X1><1>G=<X3><2>G  <X1><1>g=<X3><2>g  <X1><1>7=<X3><2>7  <X1><1>,=<X3><2>,
 68|<X1><1>H=<X3><2>H  <X1><1>h=<X3><2>h  <X1><1>8=<X3><2>8  <X1><1><=<X3><2><
 69|<X1><1>I=<X3><2>I  <X1><1>i=<X3><2>i  <X1><1>9=<X3><2>9  <X1><1>.=<X3><2>
 70|<X1><1>J=<X3><2>J  <X1><1>j=<X3><2>j  <X1><1>0=<X3><2>0  <X1><1>>=<X3><2
 71|<X1><1>K=<X3><2>K  <X1><1>k=<X3><2>k  <X1><1>!=<X3><2>!  <X1><1>/=<X3><2
 72|<X1><1>L=<X3><2>L  <X1><1>l=<X3><2>l  <X1><1>@=<X3><2>@  <X1><1>?=<X3><2>
 73|<X1><1>M=<X3><2>M  <X1><1>m=<X3><2>m  <X1><1>#=<X3><2>#  <X1><1>`=<X3><2>`
 74|<X1><1>N=<X3><2>N  <X1><1>n=<X3><2>n  <X1><1>$=<X3><2>$  <X1><1>~=<X3><2>~
 75|<X1><1>O=<X3><2>O  <X1><1>o=<X3><2>o  <X1><1>%=<X3><2>%  <X1><1>\=<X3><2>\
 76|<X1><1>P=<X3><2>P  <X1><1>p=<X3><2>p  <X1><1>^=<X3><2>^  <X1><1>|=<X3><2>|
 77|<X1><1>Q=<X3><2>Q  <X1><1>q=<X3><2>q  <X1><1>&=<X3><2>&  <X1><1>
 78|<X1><1>R=<X3><2>R  <X1><1>r=<X3><2>r  <X1><1>*=<X3><2>*  <X1><1>
 79|<X1><1>S=<X3><2>S  <X1><1>s=<X3><2>s  <X1><1>(=<X3><2>(  <X1><1>
 80|<X1><1>T=<X3><2>T  <X1><1>t=<X3><2>t  <X1><1>)=<X3><2>)  <X1><1>
 81|<X1><1>U=<X3><2>U  <X1><1>u=<X3><2>u  <X1><1>-=<X3><2>-  <X1><1>
 82|<X1><1>V=<X3><2>V  <X1><1>v=<X3><2>v  <X1><1>_=<X3><2>_  <X1><1>
 83|<X1><1>W=<X3><2>W  <X1><1>w=<X3><2>w  <X1><1>==<X3><2>=  <X1><1>
 84|<X1><1>X=<X3><2>X  <X1><1>x=<X3><2>x  <X1><1>+=<X3><2>+  <X1><1>
 85|<X1><1>Y=<X3><2>Y  <X1><1>y=<X3><2>y  <X1><1>[=<X3><2>[  <X1><1>
 86|<X1><1>Z=<X3><2>Z  <X1><1>z=<X3><2>z  <X1><1>{=<X3><2>{  <X1><1>
 87|
--------------------------------------------------------------------------------
A90
Type entry or use  commands                                          1:41 pm
```

Figure 3: KEY.REF Template with SS Data



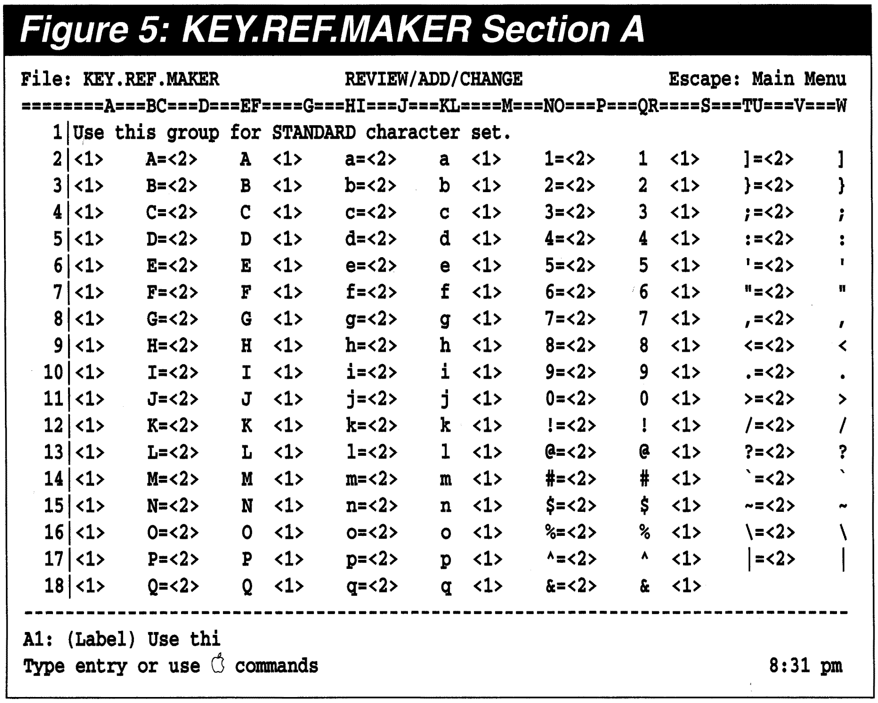
```
File: KEY.REF                    REVIEW/ADD/CHANGE              Escape: Main Menu
=====<====<====<====<====<====<====<====<====<====<====<====<====<====<===
<1=/CEFE1/APPLEWORKS.3.0/FONTS/COURIER.12>
<2=/CEFE1/APPLEWORKS.3.0/FONTS/MOBILE.18>
<1>Font:
Character Set:

<1>     A = <2>    A   <1>     a = <2>      a
<1>     B = <2>    B   <1>     b = <2>      b
<1>     C = <2>    C   <1>     c = <2>      c
<1>     D = <2>    D   <1>     d = <2>      d
<1>     E = <2>    E   <1>     e = <2>      e
<1>     F = <2>    F   <1>     f = <2>      f
<1>     G = <2>    G   <1>     g = <2>      g
<1>     H = <2>    H   <1>     h = <2>      h
<1>     I = <2>    I   <1>     i = <2>      i
<1>     J = <2>    J   <1>     j = <2>      j
<1>     K = <2>    K   <1>     k = <2>      k
<1>     L = <2>    L   <1>     l = <2>      l
<1>     M = <2>    M   <1>     m = <2>      m
<1>     N = <2>    N   <1>     n = <2>      n
<1>     O = <2>    O   <1>     o = <2>      o
-----------------------------------------------------------------3.08M Avail.
Type entry or use  commands           Line 1  Column  1      22/12/94  2:01 pm
```

Figure 4: Column Widths

Column	Widths	Column	Widths
A through W	7	K	1
F, L, R	2	N, O	1
B, C	1	Q	1
E	1	T, U	1
H, I	1	W	1

Figure 5: KEY.REF.MAKER Section A



```
File: KEY.REF.MAKER              REVIEW/ADD/CHANGE               Escape: Main Menu
========A===BC===D===EF====G===HI===J===KL====M===NO===P===QR====S===TU===V===W
   1|Use this group for STANDARD character set.
   2|<1>    A=<2>   A  <1>    a=<2>   a  <1>    1=<2>   1  <1>    ]=<2>    ]
   3|<1>    B=<2>   B  <1>    b=<2>   b  <1>    2=<2>   2  <1>    }=<2>    }
   4|<1>    C=<2>   C  <1>    c=<2>   c  <1>    3=<2>   3  <1>    ;=<2>    ;
   5|<1>    D=<2>   D  <1>    d=<2>   d  <1>    4=<2>   4  <1>    :=<2>    :
   6|<1>    E=<2>   E  <1>    e=<2>   e  <1>    5=<2>   5  <1>    '=<2>    '
   7|<1>    F=<2>   F  <1>    f=<2>   f  <1>    6=<2>   6  <1>    "=<2>    "
   8|<1>    G=<2>   G  <1>    g=<2>   g  <1>    7=<2>   7  <1>    ,=<2>    ,
   9|<1>    H=<2>   H  <1>    h=<2>   h  <1>    8=<2>   8  <1>    <=<2>    <
  10|<1>    I=<2>   I  <1>    i=<2>   i  <1>    9=<2>   9  <1>    .=<2>    .
  11|<1>    J=<2>   J  <1>    j=<2>   j  <1>    0=<2>   0  <1>    >=<2>    >
  12|<1>    K=<2>   K  <1>    k=<2>   k  <1>    !=<2>   !  <1>    /=<2>    /
  13|<1>    L=<2>   L  <1>    l=<2>   l  <1>    @=<2>   @  <1>    ?=<2>    ?
  14|<1>    M=<2>   M  <1>    m=<2>   m  <1>    #=<2>   #  <1>    `=<2>    `
  15|<1>    N=<2>   N  <1>    n=<2>   n  <1>    $=<2>   $  <1>    ~=<2>    ~
  16|<1>    O=<2>   O  <1>    o=<2>   o  <1>    %=<2>   %  <1>    \=<2>    \
  17|<1>    P=<2>   P  <1>    p=<2>   p  <1>    ^=<2>   ^  <1>    |=<2>    |
  18|<1>    Q=<2>   Q  <1>    q=<2>   q  <1>    &=<2>   &  <1>
--------------------------------------------------------------------------------
A1: (Label) Use thi
Type entry or use  commands                                              8:31 pm
```

1. Copy cells A1 through W27 to the clipboard as a "Block". *[Ed: Use Apple-. and Apple-9 to select the block quickly.]* Then copy the block into cells A30 and A60.
2. Use Apple-U to change "STANDAR" to "EXTEND#" in cells G30 and G60.
3. Use Apple-U to change "D" to "1" in cell H30 and to "2" in cell H60.
4. In cell A31, use Apple-U to insert "<X1>" before the "<1>" that is already there.
5. Copy cell A31 "Within worksheet" to cells A32 through A56.
6. Copy cells A31 through A56 to the clipboard as a "Block". Then copy the block to cells G31, M31, S31, A61, G61, M61, and S61. This changes all the "<1>" labels to "<X1><1>" in Sections B and C.
7. Use Apple-U to insert "<X2>" before the "<2>" label in cell D31.
8. Copy cell D31 "Within worksheet" to cells D32 through D56.
9. Copy cells D31 through D56 to the clipboard as a "Block". Then copy the block to cells J31, P31, and V31. This inserts the command for the first extended character set into all the appropriate cells in Section B.

Now you will edit Section C for the third character set. Continue with these steps:

10. Use Apple-U to insert "<X3>" before the "<2>" label in cell D61.
11. Copy cell D61 "Within worksheet" to cells D62 through D86.
12. Copy cells D61 through D86 to the clipboard as a "Block". Then copy the block to cells J61, P61,

and V61. That inserts the command for the second extended character set into all the appropriate cells in Section C.

13. Use Apple-B to erase cells U18 to V27, U47 to V56, and U77 to V86 as a "Block".

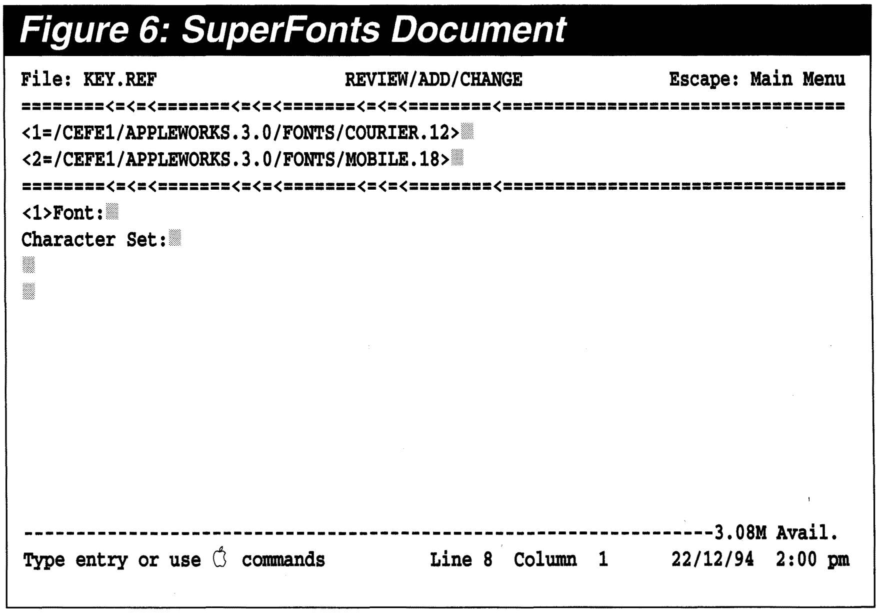
14. Print the spreadsheet. Your results should look like the example in *Figure 2*.

15. Save your template to disk.

Creating the KEY.REF Template

Now you will create the word processor document you will use to print the font preview reference sheets. Follow these steps to create the document:

Figure 6: SuperFonts Document



```
File: KEY.REF                    REVIEW/ADD/CHANGE               Escape: Main Menu
========<=<=<=======<=<=<=======<=<=<========<==================================
<1=/CEFE1/APPLEWORKS.3.0/FONTS/COURIER.12>▒
<2=/CEFE1/APPLEWORKS.3.0/FONTS/MOBILE.18>▒
========<=<=<=======<=<=<=======<=<=<========<==================================
<1>Font:▒
Character Set:▒
▒
▒









--------------------------------------------------------------------3.08M Avail.
Type entry or use  commands          Line 8  Column  1      22/12/94  2:00 pm
```

1. Create a new word processor file named "KEY.REF".

2. Insert the two SuperFonts Load Font commands in *Figure 6*. Adjust the pathnames for your system. *[Ed: You can use TimeOut PickFonts for this step.]*

Now you will set the tabs.

3. Use Apple-T to "Create new" tabs for the document. Press "N" to erase the existing tabs. Then set a "Left" tab at positions 9, 11, 13, 21, 23, 25, 33, 35, 37, 46. When you are done, press the Escape Key.

4. Press Apple-Z to display the formatting options.

5. On line 4, type "<1>Font:".

6. On line 5, type "Character Set:" and press the Return Key twice. Your screen should look like the example in *Figure 6*.

7. Save the file to disk.

Printing the Standard Character Set

The three blocks of cells in the KEY.REF.MAKER spreadsheet correspond to the three character sets provided with most SuperFonts fonts. Each section of the template contains the SuperFonts commands that you entered as labels. To print a character set, you will copy the appropriate section to the KEY.REF document. You will then use SuperFonts to print a document that contains a sample of every character in the font.

Follow these steps to print the characters:

1. Switch to the KEY.REF.MAKER spreadsheet.

2. Copy cells A2 to K27 to the clipboard as a "Block". These cells contain the SuperFonts commands for the upper- and lower-case letters of the alphabet.

3. Switch to the KEY.REF word processor document, press Apple-Z to hide the formatting options, move the cursor to line 7, and copy the contents of the clipboard into the document. Your result should look like *Figure 3*.

4. Put the cursor on line 34 and repeat steps #1 through #3 to copy the block of cells from M2 through W27 into the KEY.REF document.

5. Enter the name of the picture font next to the "Font:" label on line 4.

6. Enter the character set you are previewing next to the "Character Set:" label on line 5. For this set, enter "Standard".

7. Use Apple-N to rename the word processor document "KEY.REF.STD" and save it to disk.

8. Use SuperFonts to print the document in high-quality mode.

Alternate Character Sets

Many of the fonts you can use with SuperFonts contain as many as 255 characters, most of which you cannot access from your keyboard. Fortunately, SuperFonts' "Extra" commands (<X2> and <X3>) let you use all the pictures, letters, numbers, and symbols in the extended character sets in your documents. You use the "<X1>" command to return to the standard character set. The "Sample 7" file that came on your SuperFonts program disk demonstrates how to use the Extra command in your SuperFonts documents.

But here's a word of caution: Extended character sets are often incomplete and may contain only a few characters. However, it is worth the trouble to preview or print the complete set; it may contain just the picture you need.

Printing the Extended Character Sets

Now you should modify these procedures to print the reference sheets for the two extended character sets. Instead of copying the two blocks from Section A of the spreadsheet, copy the two blocks of Section B (A31 through K56 and M31 through W56) for Extended Set #1 or Section C (cells A61 through K86 and M61 through W86) for Extended Set #2.

In the KEY.REF document, be sure to change the "Character Set:" label to "Extended #1" or "Extended #2", depending upon which character set you are printing. When you rename the word processor document, use the names "KEY.REF.X1" and "KEY.REF.X2" for the two extended character sets.

Printing Reference Sheets for Other Fonts

You should not need to use the KEY.REF.MAKER spreadsheet again. Follow these steps to print reference sheets for another font (such as Cairo.18):

1. Open the three word processor documents: KEY.REF.STD, KEY.REF.X1, and KEY.REF.X2.
2. Edit the Load Font command on line 2 of each file to reflect the path to the new font. You will use Courier.12 to print the name of the corresponding key, so you do not need to change the reference to Courier.12 on line 1.

   For example, in this tutorial you previewed Mobile.18. You will change the pathname of font <2> when you want to print a reference sheet for another font. If the Cairo.18 font is at the same path as Mobile.18, all you need to do is substitute the new font's name for the old one.
3. Edit the "Font:" label in each document before printing the reference sheets with SuperFonts.

Conclusion

SuperFonts can help you create documents that contain interesting graphics such as the illustrations and symbols included in Mobile, Cairo, and other picture fonts. Although AppleWorks is not a what-you-see-is-what-you-get program, the procedures described in this article let you create handy reference sheets that make it easy to use these fonts in your documents. ▨

[Bill Thompson is a long-time AppleWorks user from Portland, Maine.]

[Jack Fowler is a retired Operations Manager for the Public Service Company of Colorado. Mr. Fowler has used AppleWorks since 1984.]

*[Ed: Working copies of the KEY.REF.MAKER and KEY.REF templates appear on this month's issue of **NAUG on Disk**, which costs $10 from **NAUG**. The templates require AppleWorks 2.0 or later enhanced with TimeOut SuperFonts. **NAUG on Disk** requires a 3.5-inch drive.]*

How to Convert Spreadsheet Formulas to Values

by Keith Johnson

Numbers get into spreadsheet cells in two ways. Either you type them in as data, or a formula in the cell generates the number.

There is an important difference between these two types of numbers. Although AppleWorks cannot change a number that you type into a cell, numbers calculated by AppleWorks will change automatically when any of the cells referenced in the formula changes. That's where spreadsheets get much of their power. Change the data and the formulas update your calculations.

However, that's a problem for many spreadsheet applications because AppleWorks cannot "lock in" the result of a calculation so it remains unchanged in the future. Every time you enter new data, the spreadsheet recalculates all the formulas.

Fortunately, like most of AppleWorks' shortcomings, there are work-arounds that solve this problem. Specifically, you can "lock" calculated numbers by converting them into values that AppleWorks "thinks" you typed into the spreadsheet. To do that, you replace the formula with the number generated by that formula.

AppleWorks 4.0 and later includes a macro that does the conversion. With the UltraMacros player active, you put the cursor in a cell that contains a formula, press <sa-V>, and AppleWorks will replace the formula with the number calculated by that formula. The macro works well, but only converts one cell at a time.

Dan Crutcher described three solutions to this problem in his response to a letter on page 2 of the November 1994 issue of ***AppleWorks Forum***. His methods work, but I propose a fourth approach: The set of macros in *Figure 1* that can convert formulas to values in any block of cells you highlight. The macros work with AppleWorks 3.x and later and UltraMacros 3.x and later.

How to Use the Macros

Follow these steps to enter and use the macros:

1. Type the macros into your macro file. I assigned the main macro to <ba-F>, but you can use any key combination you like. Make certain that the <asr> macro precedes the <asp> macro.
2. Compile the file and save it as your default macro set. *[Ed: Step-by-step directions for adding a macro to your default set appeared in the sidebar "How to Add a Macro" in the April 1994 issue of the* ***AppleWorks Forum****.]*
3. To convert a block of cells, press <ba-F>. A message will tell you to move the cursor to the upper left corner of the block you want to convert. Use the Arrow Keys (with the Open-Apple Key if you need it) to move the cursor. Press the Return Key when the cursor is in position. Pressing the Escape Key ends the macro. The macro will ignore any other keystrokes.
4. A message will direct you to choose the desired block of cells. Again, use the Arrow Keys along with the Open-Apple Key if desired. Then press the Return Key. The cursor must end up below and to the right of its original position. The macro will display an error message and stop if the cursor is to the left or above its original position. You can stop the macro by pressing the Escape Key here as well.

The macro will convert all the formulas in the chosen block into numbers. It then returns the cursor to its original position.

Figure 1: Macro that Converts Formulas to Values

```
<ba-F>:<asr><                   { Subroutine that moves the cursor.                                          }
 begin:                         { Begin the loop that gets the keystrokes.                                   }
  k=key:                        { Get the keystroke.                                                         }
  $1=chr$ k:                    { Convert to a string variable.                                              }
  if k=8 or k=10 or k=11 or k=21 print $1:endif:  { If it's an Arrow Key, pass it to AppleWorks.             }
  if k=136 or k=138 or k=139 or k=149 print $1:endif:  { If it's an Open-Apple Arrow Key, pass it to AppleWorks. }
  if k=27 msg "":exit:endif:    { If it's an Escape Key, erase the message and exit the subroutine.          }
  if k=13 exit:endif:rpt>!      { If it's a Return, exit the subroutine.                                     }

<ba-F>:<asp><                   { The main macro.                                                            }
 msg ' Put cursor in upper left corner, then press Return. (Esc to stop) ':  { Display these instructions.   }
 ba-F:                          { Go to the subroutine that moves the cursor.                                }
 if k=27 stop:endif:            { If the user pressed the Escape Key, stop the macro.                        }
 posn x,y:                      { Save the initial cursor position.                                          }
 msg ' Formula->Value: Choose the block you want to convert. ':  { Display these instructions.               }
 oa-C>TB<ba-F:                  { Start the "Copy block to clipboard" operation...                           }
                                { ...and go to the subroutine that moves the cursor.                         }
 if k=27 esc:stop:endif:        { If the user pressed the Escape Key, stop the copy operation.               }
 posn s,t:                      { Save the final cursor position.                                            }
 if s<x or t<y bell:            { If the cursor is above or to the left of the original position, sound beep... }
  msg ' Bad block choice! Please try again! (press a key) ':  { ...display this error message...             }
  k=key:                        { ...get a keypress from the user...                                         }
  msg "":esc:                   { ...erase the message, stop the copy operation...                           }
  endmacro:endif:               { ...and stop the macro.                                                     }
 rtn:                           { Finish the copying routine.                                                }
 msg "":                        { Erase the message.                                                         }
 oa-F>C<$1="":                  { Start routine that finds the original coordinates and initialize $1 as blank. }
 z=x-1/26:                      { Define variable z.                                                         }
 a=z*26:x=x-a:                  { Bring x into range 1-26.                                                   }
 if z>0 z=z+64:$1=chr$ z:endif: { Define $1 as the first character of the label if necessary.                }
 x=x+64:                        { Get x into range 65-90.                                               ❶    }
 $2=chr$ x:                     { Define $2 as second character of the column label.                         }
 $3=str$ y:                     { Define $3 as the row number.                                               }
 oa-Y:                          { Erase any previous coordinates.                                            }
 print $1+$2+$3:rtn:            { Print the coordinates and go there.                                        }
 oa-C>F<$0=screen 1,24,4:       { Start to copy from the clipboard.                                          }
 if $0="From">V!                { If offered the choice, choose to copy as values only.                      }
```

Technical Details

The format of some lines in the macro is unusual. For example, examine the line beginning "oa-C>TB<ba-F:". I would normally put the "ba-F" command on a new line. But you need a legitimate UltraMacros command following the "<" on the line. Rather than insert a "dummy" command in this spot, I moved the "ba-F" up from the line it would usually occupy.

Note that the last two lines in the macro test for the presence of "From" in the lower-left corner of the screen when it copies the values from the clipboard. If there are no formulas on the clipboard (because there were none in the block you selected), AppleWorks will not ask if you want to convert from formulas to values. The macro must be ready for this possibility.

There is a complicated section that re-positions the cursor (see ❶ in *Figure 1*). It uses the <oa-F> command to go to a pair of coordinates in the spreadsheet. But that position was originally stored as a pair of numbers using the <posn> command. For the <oa-F> command to work, the first coordinate must be converted into one or more letters because the spreadsheet uses letters to label its

Another Conversion Macro

Robert Boucher, of Houston, Texas, submitted the following macro which takes a different approach to performing the same formula-to-values conversion. We include it here so you can take your choice. Mr. Boucher reports that this is a modified version of a macro he found from a source he can no longer identify.

The <sa-L> command half-way through the macro ends the current macro and returns to your default macro set. Replace this command with <stop> if you typed this macro into your default set.

Figure A: Boucher's Conversion Macro

```
v:<asp><                         { Macro that converts spreadsheet formulas into values.                       }
oa-v >rfm<                       { Turn on manual recalculation to speed up the macro.                         }
msg ' Go to first cell and press Return ' :  { Get the first cell for conversion.                              }
input :                          { Get the coordinates of the first cell.                                      }
posn W, X :                      { W = column position, X = row position.                                      }
msg "" :                         { Clear the message at the bottom of the screen.                              }
msg ' Go to last cell and press Return ' :  { Get the last cell for the conversion.                            }
input :                          { Get the coordinates of the last cell.                                       }
posn Y, Z :                      { Y = column position, Z = row position.                                      }
msg "" :                         { Clear the message at the bottom of the screen.                              }
U = Ø :                          { Initialize a "flag" variable.                                               }
if W = Y then U = 1 : else :     { If the range of cells is in the same column, set the flag.                  }
if X = Z then U = 2 : endif :    { If the range of cells is in the same row, set the flag.                     }
if U = Ø then msg ' First cell and last cell are in different rows and columns ' :
                                 { The macro will not work if the cells are not in the same row or column.     }
  bell :                         { Sound the error bell...                                                     }
  sa-L :                         { ...end the macro and return to your default set.                            }
else :                           { Otherwise, the macro will work.                                             }
if U = 2 then V = Y - W + 1 :    { If the cells are in the same row, determine how many columns of cells       }
                                 { to convert.                                                                 }
  else :                         { Otherwise...                                                                }
V = Z - X + 1 : endif :          { ...the cells are in same column. Determine how many rows of cells           }
                                 { to convert.                                                                 }
begin :                          { Begin the loop that converts the cells.                                     }
  cell :                         { Read the data in the cell.                                                  }
  print $Ø : rtn :               { Print the data back into the same cell.                                     }
  V = V - 1 :                    { Reduce the counter by one.                                                  }
  if V = Ø then oa-v             { If all the cells were converted...                                          }
    >rfa<rtn :                   { ...turn auto calculation back on...                                         }
    msg "" :                     { ...clear the message at the bottom of the screen...                         }
    sa-L :                       { ...end the macro and return to your default set.                            }
  endif :                        { If all the cells were not converted...                                      }
  if U = 2 then left :           { If all cells are in the same row, move left one cell.                       }
  else :                         { Otherwise all the cells are in the same column...                           }
  up : endif :                   { ...move up one cell and...                                                  }
rpt>!                            { ...repeat the loop.                                                         }
```

columns. Moreover, the first 26 columns use a single letter, while the remaining columns use two letters. So the macro sets variable z to Ø for columns A-Z, 1 for columns AA-AZ, and so on. It then uses that value to define the second, prefixing character for the column label.

Details of Section ❶

Here is the logic behind the steps that re-position the cursor:

I originally used the <posn> command to define x as the column number of the starting cell, and y as the row number. But AppleWorks' <oa-F> command requires you to specify the column by its alphabetical label (for example, column #4 is labeled "D"; column #27 is labeled "AA"), not its number. So I must go through some gymnastics to convert the number to its equivalent alphabetical label.

I used two string variables to store the two letters for the column label. $1 will store the first letter, which will be blank if the original coordinates include columns A through Z. $2 will store the right-most letter of the column label.

To find the first letter (if any), I define **`z=x-1/26`**. UltraMacros' left-to-right arithmetic interprets x - 1/26 as (x - 1)/26. Since UltraMacros uses integer arithmetic, z is now the integer part of the formula. As a result, if 1 is in the range 1-26, the statement sets z = Ø. If x is in the range 27-52, then z = 1, and so on. Finally, if z is not zero, I add 64 to the value of z so I can use the <chr$> command to convert the result to the correct capital letter. If z = Ø, I leave $1 blank, since there will be no first letter.

Now for the second letter. The first letter controls which set of 26 columns the cell falls in, and z is the number of that set. I define **`a=26*z`**, and subtract it from x. That "throws out" sets of 26, leaving x in the range 0-26. Then I can convert the resulting number into a letter just as I did for the variable z.

Finally, I define $3 as the ASCII expression of the row number, from variable y. I do not need to convert this value into a letter, since AppleWorks labels its spreadsheet rows by number. When AppleWorks asks which cell to go to, the macro responds by concatenating $1 + $2 + $3 and entering that as the cell coordinates.

This is certainly a convoluted logic. If any **NAUG** member can think of a shorter and more elegant method for this operation, please send it in.

[Keith Johnson is Associate Director of the Fleischmann Planetarium at the University of Nevada.]

*[Working copies of Keith Johnson's and Robert Boucher's macros appear on this month's issue of **NAUG on Disk**. The macros require AppleWorks 3.0 and UltraMacros 3.x or later. **NAUG on Disk** requires a 3.5-inch disk drive.]*

Special NAUG Discount on Shareware Solutions II

NAUG members who used to read inCider magazine probably remember Joe Kohn's lively column about new Apple II public domain and shareware programs. When inCider readers asked Mr. Kohn for sources for these disks he started "Shareware Solutions", which sold the disks to readers at discount prices.

With the demise of inCider, Mr. Kohn introduced Shareware Solutions II, a bi-monthly newsletter filled with difficult-to-find information of interest to Apple II users. The latest issue of Shareware Solutions II contains a description of Mr. Kohn's interviews with Steve Wozniak and Olivier Zardini (author of the Cogito puzzle game), news about the latest Apple II events and products, four pages of interesting information about the Internet, five special offers to readers, an article that describes how to improve your AppleWorks printouts, and descriptions of new disks available from the Shareware Solutions library.

A 12-issue subscription to Shareware Solutions II usually costs $35. Until June 1, **NAUG** members in the U.S. and Canada can subscribe to Shareware Solutions II for $30, which includes first class postage. **NAUG** member subscriptions outside North America cost $45 (regularly $50) and include international airmail shipment of the newsletter.

Mr. Kohn assumes that subscribers want the complete collection of newsletters, so your subscription will include all eight back issues and the next four issues (eight months) of Shareware Solutions II. Include your check or money order in U.S. funds made payable to "Joe Kohn" with your order. Mr. Kohn does not accept charge cards, purchase orders, or COD orders.

[Joe Kohn, Shareware Solutions II, 166 Alpine Street, San Rafael, California 94901.]

How to Get Started with AppleWorks 5

by Dan Crutcher

The long-awaited package from Quality Computers arrives. You rip open the box, dump its contents onto your desk, and extract the essential items: the two 3.5-inch disks titled "AppleWorks 5 Install Disk" and "Program Disk".

But rather than plunge ahead, you hesitate. Beneath the excitement, you feel an undercurrent of fear. You recall the endless time and energy it took to upgrade from AppleWorks 3.0 to AppleWorks 4-point-whatever and the confusion that resulted – the new standard settings, the re-compiled macros, the TimeOuts that forever needed updating.

Not to worry. Installing AppleWorks 5 is a breeze compared to earlier versions of AppleWorks, especially if you already upgraded to AppleWorks 4. There are a few potential pitfalls, but this article contains the tips you need to avoid them. It's like the doctor says as he fills the needle: "This is only going to hurt a little, and you'll feel much better afterwards."

"Here are the tips you need to avoid the pitfalls."

Installation

Installing AppleWorks 5 is so simple that I won't waste space describing it. Turn to page 79 of the (excellent) Delta Manual and follow the instructions. You'll need about 600K of free disk space for a full installation to a hard disk. As with any new program, you should make copies of your original disks and use the copies to make any changes.

If you run AppleWorks from a 3.5-inch disk, all you need do is copy the Program Disk and you're ready to go. If you run from 5.25-inch floppies, it's time to upgrade your hardware. This version of AppleWorks requires at least one 3.5-inch disk drive or a hard disk.

Don't worry about the installer overwriting or changing any of your earlier copies of AppleWorks. It won't. In fact, it's a good idea to keep your old version handy (and installed on your hard disk if you have one) until you're comfortable with AppleWorks 5.

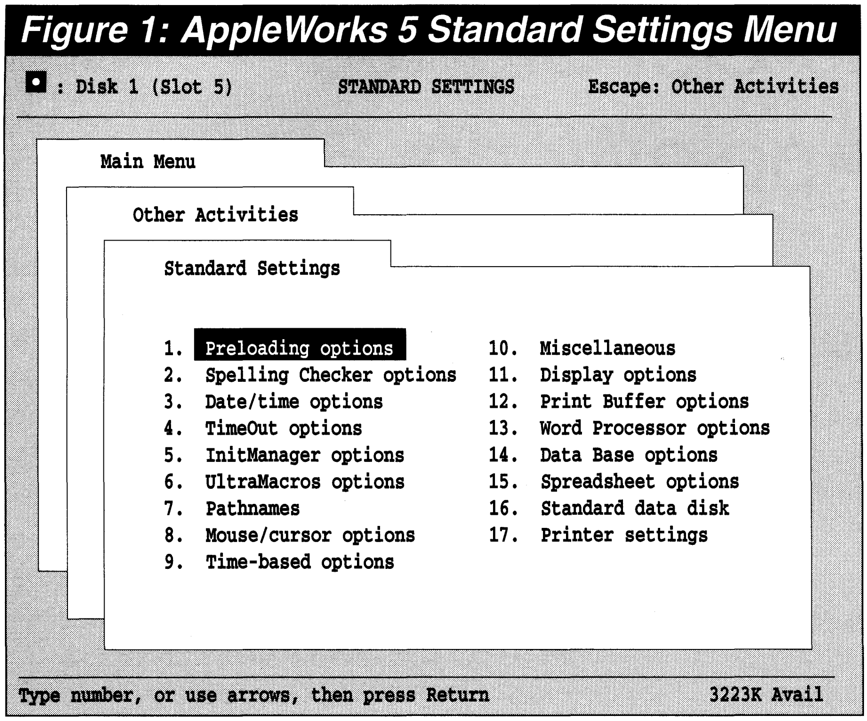
If you use a hard disk, the installer will create a subdirectory called "AW5" in the root directory of the partition you select and copy all the AppleWorks 5 files into that subdirectory. Even if you want AppleWorks 5 to reside in another directory, I recommend letting the installer put it in the root directory. After it's installed, you can use a file utility program to copy the AppleWorks 5 directory and its contents to your preferred location.

Your First Launch

The first time you launch AppleWorks 5, the program will display a screen that asks you to type your name and address. Follow those directions.

If you moved the AW5 directory, AppleWorks will display a message telling you that it cannot find the AW.INITS file. (AppleWorks displays this message because the AppleWorks 5 installer sets the pathname to the AW.INITS and TIMEOUT directories to the directory where you originally installed AppleWorks 5. You must change that information if you move AppleWorks 5 to another directory. I'll show you how in the next section.) If that happens, re-launch AppleWorks and press the Escape Key as many times as necessary to allow AppleWorks 5 to load without any Inits or TimeOut applications.

Figure 1: AppleWorks 5 Standard Settings Menu



```
■ : Disk 1 (Slot 5)          STANDARD SETTINGS          Escape: Other Activities

    Main Menu
        Other Activities
            Standard Settings

            1. Preloading options           10. Miscellaneous
            2. Spelling Checker options     11. Display options
            3. Date/time options            12. Print Buffer options
            4. TimeOut options              13. Word Processor options
            5. InitManager options          14. Data Base options
            6. UltraMacros options          15. Spreadsheet options
            7. Pathnames                    16. Standard data disk
            8. Mouse/cursor options         17. Printer settings
            9. Time-based options

Type number, or use arrows, then press Return                        3223K Avail
```

If you haven't moved AppleWorks 5, fill in your name and address at the screen prompts. When you complete that process, AppleWorks 5 will create a file called NAME.ADDRESS in your AW 5 directory. You won't see that screen again when you start AppleWorks, but you will be able to use a macro that automatically inserts your name and address in a word processor file. Try it: Create a new word processor file and press <sa-B> to start a memo.

To change the information in NAME.ADDRESS, just delete that file from your AW 5 directory. The next time you launch AppleWorks the program will prompt you for new name and address entries.

Standard Settings

The Standard Settings Menu in *Figure 1* is the first place you should visit the first time you run AppleWorks 5. (The quickest way to get there is to press <oa-Q>, then <oa-S>.) As you can see from *Figure 1*, AppleWorks 5 lets you control many more settings, including some options that used to be available only through third-party "patch" programs.

Appendix B in AppleWorks 5's Delta Manual documents all the new features in the Standard Settings Menu; you should spend some time exploring these options. Here are a few of the key settings and some changes you might consider. (The numbers in the following section refer to the item numbers for these options in the Standard Settings Menu.)

1. Preloading Options

If you run AppleWorks from a hard disk or a RAM disk, try setting preloading to "None." If you run AppleWorks 5 from floppies and you have one megabyte or more of memory, leave the preloading options set to "All".

2. Spelling Checker Options

If you moved the AW5 directory from its original location, you need to use "Location of directories" (item #7 in the Spelling Checker Options Menu) to tell AppleWorks where to find your dictionaries.

4. TimeOut Options

If you moved the AW5 directory, you need to use item #4, "Timeout Directory" from the InitManager Options Menu, to specify the new location of your TimeOut applications. For example, I changed the pathname to my TimeOut directory to /HD/PROGRAMS/AW5/TIMEOUT.

5. InitManager Options

If you moved the AW5 directory, select #2, "Init Directory", and enter the pathname to your inits. For example, I changed the pathname to /HD/PROGRAMS/AW5/AW.INITS".

6. UltraMacros Options

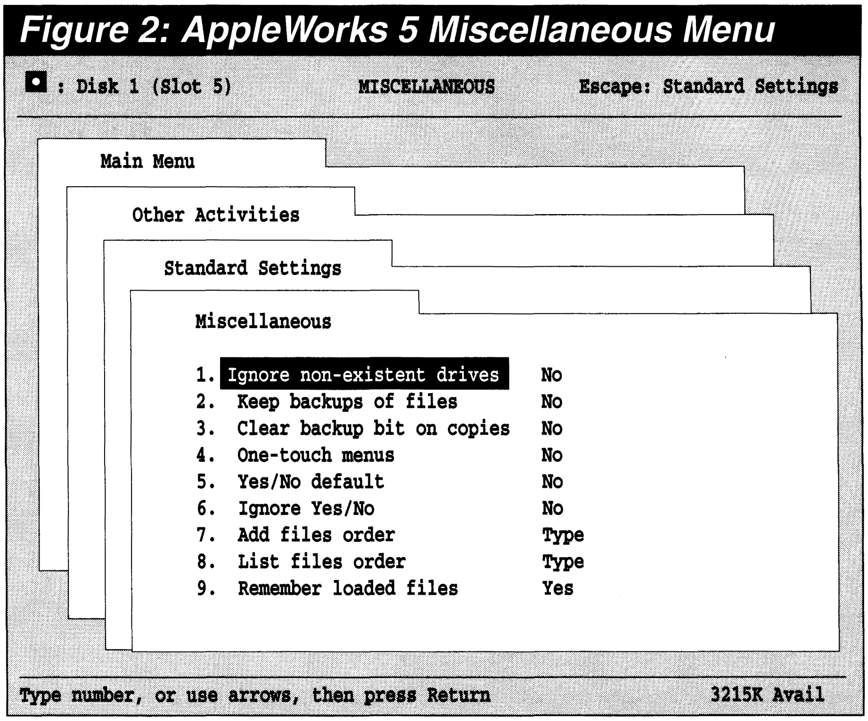
This is where you turn UltraMacros on or off. There is no reason to turn UltraMacros off unless you experience problems with AppleWorks crashing. (The crashes are usually caused by conflicts between UltraMacros and some GS/OS applications that use interrupts. Launching AppleWorks 5 from ProSel-16 appears to cure this problem.)

AppleTalk is one interrupt-intensive application that can cause crashes. If AppleTalk is active when you launch AppleWorks 5, AppleWorks 5 disables UltraMacros and displays the message “Macros cannot be used with AppleTalk active.” (See the section entitled “Randy’s Free Patcher” at the end of this article to learn how to activate UltraMacros while AppleTalk is active.)

8. Mouse/Cursor Options

You need to speed up the cursor blink rate if you enable the mouse. For a stock Apple IIe (which runs at 1 megahertz), try setting both the “cursor on” and “cursor off” rates to “2”. For a stock Apple IIGS (which runs at 2.8 megahertz), try setting these values to “5”. For a IIGS running at 8 megahertz, set the values to “10” to get the normal cursor blink rate.

Figure 2: AppleWorks 5 Miscellaneous Menu



```
■ : Disk 1 (Slot 5)          MISCELLANEOUS          Escape: Standard Settings

  Main Menu
    Other Activities
      Standard Settings
        Miscellaneous

          1. Ignore non-existent drives    No
          2. Keep backups of files         No
          3. Clear backup bit on copies    No
          4. One-touch menus               No
          5. Yes/No default                No
          6. Ignore Yes/No                 No
          7. Add files order               Type
          8. List files order              Type
          9. Remember loaded files         Yes

Type number, or use arrows, then press Return                  3215K Avail
```

If you ever disable the mouse, remember that the default settings are 176 for “cursor on” and 160 for “cursor off”.

10. Miscellaneous

Figure 2 depicts AppleWorks 5’s Miscellaneous Menu. The most notable changes that you control from this menu include the following: One-touch menus no longer interfere with macros. (Give this feature a try. Change the setting in the Miscellaneous Menu and then press the number you want to choose when AppleWorks displays a menu.) The option to switch AppleWorks’ “No/Yes” queries to “Yes/No”, which used to require a patch program, is now located here. It’s called the “Yes/No default”.

12. Print Buffer

AppleWorks 5 offers a built-in print buffer, but it’s a two-edged sword: Activating the buffer lets you get back to AppleWorks faster during a sizable print job, but it slows down the actual printing. In addition, AppleWorks is often sluggish while the buffer is printing.

I experienced lock-ups while using the print buffer and heard that others had similar problems, so use this feature with caution.

16. Standard Data Disk

The default standard disk is “/EXTRAS”; you should change this setting to the disk drive or pathname you use to store your AppleWorks data.

17. Printer Settings

As with every upgrade of AppleWorks, you must re-install your printers in AppleWorks 5. The SEG.ER file (which contains your printer settings) from earlier versions of AppleWorks will not work with AppleWorks 5, so you will have to add your printer from the list of printers or define a custom printer.

If you have an ImageWriter II, you should remove the installed ImageWriter II and add a new ImageWriter II printer from the AppleWorks 5 list of printers. The ImageWriter II driver that comes installed in AppleWorks 5 has an interface setting of “Control-I 8ØN Escape c”, which is incorrect (it should be “Control-I 8ØN”). Removing and reinstalling the ImageWriter II driver will install the correct interface card setting.

And if you use a parallel interface for printing, don’t forget to change the interface card setting to whatever you used for earlier versions of AppleWorks. A setting of “Control-I ØN” or “None” works with most parallel interface cards.

*[Ed: AppleWorks 5 owners who installed a custom printer on an earlier version of AppleWorks should order **NAUG's** AppleWorks 5 Printer Drivers disk. That disk contains (a) the printer drivers for more than 100 printers not normally supported by AppleWorks, and (b) a program that converts your AppleWorks 3 and AppleWorks 4 printer drivers for use with AppleWorks 5. The AppleWorks 5 Printer Drivers disk costs $6 (plus $2 s/h) from **NAUG.**]*

Updating Macros and TimeOut Modules

When you launch AppleWorks 5, the default macro set displays the message "Default Macros Successfully Installed – Press any key." If you never used UltraMacros before, you should keep the default set intact for now. Then read the information in the file "AppleWorks 5 Macros" in the /EXTRAS /MACROS directory on the AppleWorks 5 Install Disk. That file contains descriptions of all the default set macros. *[Ed: An updated version of this file appears on this month's NAUG on Disk.]*

The default set source code is in the files "SEG.UM.source" and "SEG.AX.source" in the same directory. SEG.UM is the actual default set; SEG.AX contains macros that are used by SEG.UM.

However, if you lovingly created your own custom default macro set, you'll probably want to make that the default AppleWorks 5 set. Here's how:

1. Load the word processor file that contains your default macro set source code from your previous version of AppleWorks. (If you never created such a file, you probably don't have a custom default set. In that case, you should skip this section and use the default set that comes with AppleWorks 5.)

2. Use AppleWorks 5's TimeOut Ultra Compiler to compile your custom default set. If you are upgrading from AppleWorks 4, you should have no problem compiling the macros.

If you are upgrading from AppleWorks 3 or an earlier version of AppleWorks, you may encounter some errors while compiling. Newer versions of UltraMacros eliminated some macro commands and added others. You will need to familiarize yourself with those changes to eliminate the errors. Good sources for up-to-date macro information are the A2 Roundtable on GEnie and the **NAUG** BBS. The best published reference book for UltraMacros 4 is Will Nelken's "Ultra – to the Max" (Marin MacroWorks, 1675 Grand Avenue, San Rafael, CA 94901. GEnie e-mail address: W.NELKEN1; Internet: w.nelken1@genie.geis.com. "Ultra – to the Max" usually sells for $28.95. Until June 1, NAUG members can order this excellent reference book directly from the publisher for $25.50, which includes postage.)

The most common macro error encountered when converting from UM 3.1 to UM 4.x is the need to replace the <elseoff> token with <endif>. UltraMacros 4.x no longer supports the <elseoff> token.

3. Use TimeOut Ultra Options and choose menu item #4, "(Save current macros as) the default set." Answer "Yes" to the "Activate auto startup?" query.

Now that you updated your default set, try a few of your macros to make sure they're properly installed. Keep in mind that any macros that use <launch>, <call>, or <link> won't work until you recompile and re-create the task files they are trying to launch, call, or link to. Which means it's time to update your macro task files.

Macro Task Files

You must recompile your AppleWorks task files before you can use them with AppleWorks 5; you can't just copy the old task files into your AppleWorks 5 directory and expect them to work. Follow these steps to recompile your task files:

1. If you have the source code for the task file, add it to the desktop and use Ultra Compiler to compile it.

2. Use Ultra Options, menu item #3, "(Save current macros as) a task file."

If you assigned a name to the task file (under the "labels" section of your source code), it will appear on the bottom line of the screen; accept that as the name.

If you did not assign a name, Ultra Options will display the default name "michael", which you should rename. If you launch, call, or link to this task file from another set, be sure to give it the name you used in those “triggering” macros.

3. After you accept a name for the task file, Ultra Options will ask if you want to create a “hidden” task file. In most cases you should answer “No.” Hidden task files have a filetype of BIN, which means they won’t show up in the list of launchable task files when you use Ultra Options “Launch a Task File” option, nor can you launch them from outside AppleWorks.

Adapting Your Macros to AppleWorks 5

There are relatively few changes in the structure of AppleWorks that are likely to affect the operations of your macros, but here are two you should consider:

- The words are changed on some menu screens in AppleWorks 5. For example, a macro I use often that automatically adds files to the desktop wouldn’t work properly in AppleWorks 5. I traced the problem to a word change in the menu that lists the available files. Where it used to say “AppleWorks Files” at the top of that filecard it now says only “Files”. (That more accurately reflects AppleWorks 5’s ability to display ASCII text files as well as AppleWorks files.) My macro expected to find “AppleWorks” at a certain location on that screen, so I had to change it to look for the word “Files” instead.
- There are significant changes in AppleWorks 5’s Standard Settings Menu. You might have to revise any macros that select these settings by menu item number; AppleWorks 5 added many options to the list and moved others.

TimeOut Applications

AppleWorks 5 comes with 21 TimeOut applications, most of which you probably won’t use regularly (if at all). Here’s a quick list of some you might consider deleting or moving from the TimeOut directory of your working disk (not from the original disk):

DIF TO DB and **DIF TO SS** transfer DIF format files into AppleWorks. Most AppleWorks users will never encounter such files.

Helpful Task File Macros

UltraMacros task files are easier to manage if you use macros to compile and save the files. Here are three macros that I include in every task file set I write:

Macro 1: Compile the current word processor file.

```
<ba-C>:<awp $0 = "Ultra Compiler" : oa-esc : find : if z =
0 then msg $0 + " not found" : bell stop endif rtn oa-rtn>!
```

Macro 2: Save the current macro set as a task file; use the labels name.

```
<sa-ctrl-T>:<awp $0 = "Ultra Options": oa-esc: find: if z =
0: msg $0 + " not found ": stop else rtn print "3": rtn
rtn>NY<k=key: esc esc>!
```

This macro creates a non-hidden task file that automatically replaces any task file that has the same name. Thus, each of your task files must have a unique name. To name a task file, type “labels” (without quotes) at the top of your macros (on a line above the word “start”). Then, on a line between “labels” and “start”, type your task file name, preceded by a period. It should look something like this:

```
labels
.SAMPLE.TASK
start
```

You can define other labels in that section, but at a minimum you should have a task file name. I like to append the suffix “.TASK” to all my task files so I can easily identify them when I catalog my AppleWorks directory.

Macro 3: Launch the default macros.

```
<ba-L>:<all launch "seg.um">!
```

<ba-L> is the standard keystroke used by TAPL programmers to launch default macros.

Once you have these macros in your task file macro set, you can change your macros and update your task files by pressing <ba-C>, then <sa-ctrl-T>, without launching the TimeOut Ultra Compiler or Ultra Options.

When you re-compile your task files, save them back to disk. I keep the different versions straight by appending a “4” or “5” to the source code file name before saving it to disk.

If You Don't Have the Source File

You may not have the word processor source files for some of your task files. If that happens, try to contact the author or publisher of the macros to obtain AppleWorks 5-compatible versions of these files. Try these techniques if you cannot get an AppleWorks 5-compatible version of a task file:

- Activate the task file in AppleWorks 4 and press the Escape Key repeatedly to try to stop it without re-launching your default macros. Press Apple-Control-X (Apple-Clear on an Apple IIGS) to try to access Debug. Then try using Debug's <oa-N> command to stop the macro.
- Create a blank word processor file and use Ultra Compiler's "Display current macro set" to display the macro source code. Save the source code file to disk. Then launch AppleWorks 5, load the source code file, and try to re-compile and save the task file.

This procedure won't work with all task files, but it's worth a try. However, some task files will automatically restore the default macros when you press the Escape Key or otherwise try to stop them. In that case, you will have to contact the author.

DATE.RESET and **SCREEN.BLANKER** are for clockless computers. Delete them if you have a clock.

IWIIFIX corrects an obscure database printing problem that occurs only on ImageWriter IIs with a built-in print buffer. If you don't have such a printer, you can delete this file. Even if you do, you will probably use it only once.

INSTALL.DESKJET is useful only if you have a Hewlett-Packard DeskJet printer.

PICKFONTS is useful only if you use Super-Fonts.

The only TimeOut module you absolutely must keep (assuming you want to use TimeOut applications at all) is UTILITIES, although most of us will also want to keep Ultra Compiler, Ultra Options and Ultra Mac2Menu. Try the others out and decide for yourself which ones are useful to you. But I suggest that you keep DB REPLACE, which gives you find/replace capabilities in database files.

To add TimeOut applications from earlier versions of AppleWorks, read and follow the instructions in the file "Updating TimeOut" in the /EXTRAS /UPDATER directory.

Keep in mind that "user-created" TimeOut applications will not be updated by the TimeOut updater. These are actually task files that were converted into TimeOut applications with the Macros-to-Menus TimeOut application. To update these applications for AppleWorks 5, you must re-compile the macro source file and use Ultra Mac2Menu to create the TimeOut application. (Then they won't show up until the next time you launch AppleWorks 5.)

If you don't have the source code for these user-created TimeOut applications, you will have to contact the author, or you can try the method described in the sidebar entitled "If You Don't Have the Source File" for re-creating task files without the source code.

Randy's Free Patcher

The venerable tradition of "patching" AppleWorks – altering the way the program works to suit individual preferences – is continued in the latest version of Randy Brandt's free patch program on the AppleWorks 5 Install Disk. The file "RFP.docs" on that disk describes the patches.

To use the patcher, boot the Install Disk and choose "Randy's Free Patcher" from the first menu, or launch the BASIC program "RFP" from any program launcher.

The only problem I encountered with Randy's patch program was the "AppleTalk macros on/off" option, which I could not get to work. This option should let UltraMacros work while AppleTalk is active (on some computers running UltraMacros with AppleTalk active will cause AppleWorks to crash, which is why the default setting is "off").

I was able to use UltraMacros with AppleTalk active in AppleWorks 4, so I used RFP to turn this option on. However, I still got the message "Macros cannot be used with AppleTalk active" when I launched AppleWorks 5 and chose "UltraMacros options" in the Standard Settings menu. I finally had to use a block editor to make the change manually.

If you know how to use a block editor (such as ProSel-8's Block Warden or ProSel-16's Zap), you need to change the byte at offset +148F in APLWORKS.SYSTEM. Change the value from "EE" to "2C" to enable UltraMacros while AppleTalk is active. Note that this location differs from its location in AppleWorks 4.

Conclusion

If you followed these tips and instructions, your copy of AppleWorks 5 should be ready to use. With all the capabilities and quirks of your individual preferences and custom macros restored, you are ready to keep AppleWorking into the next millennium. ▩

[Dan Crutcher is a magazine publisher who lives in Louisville, Kentucky. He is the author of CheckWorks, TCXpress, TXC/NAUG, and other UltraMacros-based programs.]

AppleWorks News

ICON Stops Publications

ICON, the umbrella organization that publishes A2 Central, TimeOut Central, Script Central, and other disk-based publications, is going out of business. At press time, plans for the future of ICON's Apple II publications remain uncertain. ICON hopes that some will be continued by their editors or by other organizations.

ICON is preparing a letter that will notify subscribers of the cessation of its publications and its plans for the future. **NAUG** will publish ICON's plans and those for Randy Brandt's popular TimeOut Central disks in next month's issue of the ***AppleWorks Forum***.

Special Discounts from NAUG

Back Issues

Each issue of the ***AppleWorks Forum*** is filled with tips, techniques, and ideas to help you get more from AppleWorks. Now is the time to fill in the gaps in your library of back issues.

Back issues of the ***AppleWorks Forum*** normally cost $4 per copy. Until June 1, 1995, **NAUG** members can buy the complete collection of 84 available back issues for $50 plus $10 s/h. This reduces the price of back issues to less than 75 cents per copy and saves you more than $250 on the cost of these issues.

NAUG accepts Visa and MasterCard. International orders by credit card only; international shipping additional.

*[Back Issues Special, **NAUG**, Box 87453, Canton, MI 48187; (313) 454-1115; Fax: (313) 454-1965.]*

1040Works

Nobody can make preparing your tax forms fun, but 1040Works can certainly make it easier.

1040Works is a set of professionally-developed AppleWorks templates that prepare the 23 most widely-used Federal Income Tax forms. You enter your data in the templates, press Apple-K, and AppleWorks does all the work. Last minute deductions are no problem with 1040Works. Change your entries in the template, press Apple-K, and 1040Works updates your forms.

1040Works costs $29.95 plus $3.50 s/h directly from NAUG. (Non-**NAUG** members: Add $3.) The 1040Works Tax Planner, which calculates your quarterly estimated tax payments and withholding and helps you develop effective tax-saving strategies, costs $29.95 plus $3.50 s/h ($19.95 including shipping if ordered with 1040Works).

*[**NAUG**, Box 87453, Canton, Michigan 48187; (313) 454-1115; Fax: (313) 454-1965.]* ▩

Why Isn't the BBS a Toll-Free Call?

Dear **NAUG**:

I appreciate all the help I receive on **NAUG**'s AppleWorks bulletin board service. But I wonder; in today's age of toll free telephone calls, why don't we have 800 number access to the BBS?

Jeanie Slaugenhoup
Manchester, Massachusetts

[Ed: Although 800 numbers are toll free to the caller, the receiver ends up paying for the call. Since the BBS is a free service, we would have to collect the 800-number fees from the BBS users.

The cheapest 800 number service we can find would charge us 26 cents per minute daytime and 15 cents per minute evenings and weekends. Most of our members use long distance discount services that let them call the BBS for as little as 19 cents a minute daytime and 10 cents per minute evenings and weekends. So the members save money by paying their bill directly instead of our charging callers to use an 800 number.]

How to Convert Formulas to Values

Dear **NAUG**:

A letter in the November issue of the ***AppleWorks Forum*** asked how to convert formulas to values in AppleWorks 4. AppleWorks 4.x and 5.0 can do the conversion; just use sa-V in the default macro set.

Steve Beville
Spartanburg, SC

[Ed: Steve Beville should know. He is the author of most of the macros supplied with AppleWorks 4.x and 5.0.

*This is one of a number of responses we received to the formulas-to-values letter in the November issue. Keith Johnson and Robert Boucher submitted more sophisticated macros that do the conversion; their responses appear on pages 7-10 in this month's issue of the **AppleWorks Forum**.]*

Printing NLQ Output from AW 5

Dear **NAUG**:

I need some help printing near letter quality (NLQ) output on my ImageWriter II using AppleWorks 5. I use the buttons on the ImageWriter to set the printer to NLQ mode. But when I print, AppleWorks 5 turns off the NLQ lights and resets the printer to draft mode. This didn't happen with earlier versions of AppleWorks but occurs whether or not I use the printer buffer with AppleWorks 5.

Does anyone know why this occurs and how to correct the problem?

John Morgan
Florissant, Missouri

[Dan Crutcher replies: AppleWorks 5 includes the wrong printer interface card codes. The program shipped with the code Control-I 8ØN Escape c. Escape c is the code that resets the defaults on the ImageWriter; that code is turning off your NLQ setting.

To fix the problem, change the ImageWriter interface card code to Control-I 8ØN in your copy of AppleWorks 5.]

The ***National AppleWorks Users Group*** **(NAUG)** is an association dedicated to supporting AppleWorks users. **NAUG** provides technical support and information about AppleWorks and enhancements to that program. Our primary means of communicating with members is through the monthly newsletter entitled the ***AppleWorks Forum***.

AppleWorks Forum

Editor: Cathleen Merritt
Associate Editor: Warren Williams
Contributing Editor: Cynthia Field
Page Layout: Nanette Luoma
Publisher: The National AppleWorks Users Group

*©COPYRIGHT 1995, by **NAUG**, The **National AppleWorks Users Group**, for the exclusive use and enjoyment of its members. Any reprint or reproduction must be approved in writing and in advance by **NAUG**.*

*The "**AppleWorks Forum**" (ISSN 0893-4118) is published ten times annually for $30 per year by the National AppleWorks Users Group, 49068 Harvest Dr., Plymouth, MI 48170.*
Second Class postage paid at Plymouth, MI, and additional mailing offices.
POSTMASTER: Send address changes to AppleWorks Forum, NAUG, Box 87453, Canton, MI 48187

Easier Patches for AppleWorks 4

Dear **NAUG**:

Here are suggestions that might improve some of the procedures described in the November 1994 issue of the ***AppleWorks Forum***:

"How to Organize 5.25-inch Disks" (page 12): Most 5.25-inch disk users work on Apple IIe and IIc systems. These users can save space on their startup disk by deleting SEG.RM (it's the memory manager for the Apple IIGS), SEG.SN, and SEG.DR (which are parts of the data base and spreadsheet modules respectively). That saves more than 60 blocks you can use for SEG.IM and your favorite inits, and SEG.UM and the UltraMacros-related files.

"A Spell Checker Patch for AppleWorks 4" (page 18): An easier approach to changing the size of the suggested word list is to create an init that patches AppleWorks after you launch the program. Using an init lets you disable the patch without editing the file; just move the init out of your inits subdirectory and re-start AppleWorks.

Follow these steps to create the init:

1. Quit AppleWorks, launch BASIC.SYSTEM, and type "CALL -151" to access the Monitor.
2. Type the following, with a Return at the end of each line:

```
4000: 4C 15 40 6D 62 0A 0E 44 61 6E 27 73 20 53 43 20
4010: 50 61 74 63 68 20 06 30 1F 40 2E 00 19 00 60 00
4020: 00 20 48 11 0A 80 15 BB 12 00 20 48 11 FC A4 27
4030: BB 07 00 60 04 51 75 69 74 A2 03 C9 06 B0 06 CA
4040: C9 04 B0 01 CA 60 20 0F 80 EA EA EA EA
3D0G
```

3. Save the file in your INITS folder by typing:

```
BSAVE xxx/I.DANS.SC.PATCH,A$4000,L$4D
(replace xxx with the pathname to your inits)
```

The $Ø3 value that the author suggests generated too many words for my needs, so I changed it to $Ø2 to get more acceptable word lists. To make this change, launch BASIC.SYSTEM and type

```
POKE 768,2
BSAVE xxx/I.DANS.SC.PATCH,A$768,L1,B$3A
(replace xxx with the pathname to your inits)
```

"How to Avoid AppleWorks 4 Crashes on IIGS System" (page 20): The approach suggested in this article works, but it's a lot easier to patch AppleWorks with BASIC. Follow these steps:

1. Launch BASIC.
2. Type the following:

```
PREFIX /xxx  (replace "xxx" with the pathname to
                  APPLEWORKS.SYSTEM)
POKE 768,44
BSAVE APLWORKS.SYSTEM,a$300,l1,b$142B
```

Unfortunately, you cannot patch this portion of AppleWorks with an init; this segment of code runs before AppleWorks calls its inits.

Chris Serreau
Angers, France

AppleWorks 5 Inits and the Apple IIe

Dear **NAUG**:

Do you know which AppleWorks 5 inits I need on my Apple IIe system? Do I need to keep the Apple IIGS commands? Finally, what are the "Goodies" and what do they do?

And while I'm on the subject, is there any documentation for the complete set of default macros that came with AppleWorks 5 other than the information on the accompanying word processor file? Some of the macros don't seem to be of much use, but I'm afraid that I'm missing something important.

John Morgan
Florissant, Missouri

[Steve Beville replies: You don't need the IIGS commands on an Apple IIe. Those are specifically for the IIGS and don't do anything on a IIe except take up memory.

Look on the /EXTRAS disk (the label calls it the "Install Disk") in the UltraMacros subdirectory for the file "Dot.Goodies". It tells you about that set of dot commands.

Finally, the documentation for the default macro set is in a file called "AW5 Macros" in the

Macros subdirectory on the /EXTRAS disk. Unfortunately, the file isn't quite up to date. I uploaded a corrected version to the **NAUG** *BBS library that you can look at. (Ed: Search for the file "AW5MACRO.BXY" in the library. The file also appears on this month's issue of* ***NAUG on Disk*** *which costs $10 from* **NAUG**.*)*

While I'm thinking about it, someone asked me the other day why the <sa-ctrl-r> macro (which reverts to last saved version of a file) only works in the word processor. It should be a global (<all>) macro but I typed <awp> by mistake. (It has probably been like this since AppleWorks 4.0, because I don't think I looked at the macro since I wrote it.) To correct the problem, load the file SEG.AX.source on your AppleWorks desktop and change "awp" to "all" for that macro. Then recompile the file and use UltraOptions to create a hidden task file. Finally, save the changed source file on your backup copy of the /EXTRAS disk.]

How to Update Multiple Records

Dear Cathleen:

I need to replace some data in one of my large AppleWorks data base files. Are there any add-ons that give AppleWorks' data base module an equivalent of the Replace command built into AppleWorks' word processing module?

Thomas Woodell
Houston, Texas

[Ed: If you use AppleWorks 5, you're in luck. TimeOut DB.REPLACE that comes with AppleWorks 5 makes it easy to replace data in a data base file.

If you use AppleWorks 3.0 or AppleWorks 4.x, you can use Roy Barrows' Fill Category utility to replace information in a data base file. An AppleWorks 3.0-compatible version of Fill Category is part of ADB.UTILS on **NAUG***'s Barrows Utilities #4 disk. An AppleWorks 4.x version of the utility appears in the Tools.DB file on* **NAUG***'s Barrows Utilities #16 disk.*

If you use AppleWorks 1.x or 2.x, you can update your records manually. Follow these steps:

1. *Use the data base's Record Selection Rules (Apple-R) Command to select the records you want to change. Display them in multiple record layout.*
2. *Change the entry in the first record and press the Return Key. The cursor should move to the same field in the next record. (If it doesn't, press Apple-L, the Return Key, change something in the layout, change it back, press the Escape Key, and select "Down" from the list of options that appear when you press Return.)*
3. *With the cursor in the second record, press Apple-' to copy the entry from the previous record into the new record.*

Then hold down the Apple-' keys and AppleWorks will update all the selected records.]

QRAM Test Disk Limits

Thanks for publishing Doug Gum's step-by-step article that described the different ways to create a RAM disk on Apple IIe and IIc computers. (See "How to Set Up a RAM Disk on an Apple IIe or IIc" in the December 1994 issue of the ***AppleWorks Forum***.) If you have more than a megabyte of RAM on the auxiliary slot card in your computer, you need to know this about the QRAM Test Disk program that Doug described in that article:

The QRAM software can only recognize one megabyte of memory on an auxiliary slot card. That restricts the size of the RAM disk you can create with the software. Fortunately, all the AppleWorks dictionaries easily fit on that "disk", so the one-megabyte limitation is not a problem for most RAM disk users. You can still use MEM.SYSTEM to reserve the remaining memory on the card for your AppleWorks desktop.

Howard Katz
Batavia, Illinois

How to Download NAUG's Files from America Online

by Joe Connelly

NAUG members know that they can no longer access America Online (AOL) with an Apple II computer. However, the **NAUG** area is still active on this popular on-line service, although it is visited mostly by **NAUG** members who also own Macintosh systems.

NAUG continues to maintain its comprehensive public domain library on AOL. And once you know a few tricks, you can easily download **NAUG**'s AppleWorks files with your Macintosh and use them on your Apple II.

What You Need

You need version 2.5.1 of AOL's Macintosh software, Apple File Exchange or PC Exchange 2.0 for the Macintosh, and an Apple IIGS computer running ShrinkIt-GS to follow the procedures I describe in this article. *[Ed: ShrinkIt-GS can handle the resource and data forks the Macintosh creates in the file. Standard 8-bit ShrinkIt cannot unshrink these files.]*

Apple File Exchange came with most early Macintosh computers and is available on **NAUG**'s Macintosh Transfer disk, which costs $6 (plus $2 s/h *per order*) from the **NAUG** Public Domain Library.

PC Exchange 2.0 is a system extension from Apple Computer that comes with System 7.5.

ShrinkIt GS is available from the **NAUG** Public Domain Library and costs $6 (plus $2 s/h *per order*).

Step-By-Step Procedures

Follow these steps to download an Apple II file from the **NAUG** area:

1. Log onto AOL with your Macintosh, press ⌘-K, and enter the keyword "Software".
2. Scroll to the bottom of the Software Center Menu and select "Apple II Software Center".
3. Click on "Open" to select "QuickFinder File Search".
4. Enter "NAUG" as the search criterion and click on "List Matching Files".
5. If the file you want appears on the list, download the file. If the file does not appear, click on "List More Files" until the file appears on the scroll-down menu. Then download the file and log off AOL.
6. Still working on your Macintosh, use Apple File Exchange or PC Exchange 2.0 to copy the file onto a ProDOS disk.

   If you use Apple File Exchange, make certain that "default translation" appears when you transfer the file. If it does not, de-select "AppleWorks to Microsoft Works" from the Mac to ProDOS Menu and once again transfer the file onto the ProDOS disk.
7. Launch ShrinkIt-GS on your Apple IIGS and unshrink the file. Then use it as you would any file you downloaded with your Apple II.

Conclusion

I certainly don't condone AOL's discontinued support for its Apple II software. But it's nice to know that you can still access **NAUG**'s AppleWorks library without having to change to a different on-line system. ▨

*[Joe Connelly, who is **NAUG**'s representative on America Online, has been an active **NAUG** supporter since its formation in 1986. Mr. Connelly is "NAUGjoec" on AOL.]*

New Disks in the NAUG Public Domain Library

Barrows' Utilities Disks

Roy Barrows, the prolific developer of macro-based AppleWorks enhancements, reports that the macros he developed to run under AppleWorks 4 will run under AppleWorks 5. All you need do is use the UltraMacros compiler that ships with AppleWorks 5 to re-compile the source file provided for each macro. Mr. Barrows' step-by-step directions will appear in next month's issue of the *AppleWorks Forum*.

Please notify Mr. Barrows if any of his AppleWorks 4 utilities do not run under AppleWorks 5. You can write to Mr. Barrows at 73 East Street, Sharon, Connecticut 06069.

Barrows' Utilities – Disk 16

Roy Barrows recently released his sixteenth disk filled with useful utilities for AppleWorks 4 and AppleWorks 5. The utilities on Disk 16 include:

BoxDraw: Draws a box of any size you specify.

Calc.SP: Computes the total or average of the values in any range of cells you specify.

Data.Date: Adds the dates or days of the week to any data base that contains 365 or 366 records. Makes it easy to use AppleWorks to create calendars, appointment schedules, and to-do lists.

File.Check: Checks the new/changed/unchanged/saved status of all files on your three AppleWorks desktops. Particularly useful if you fast-quit AppleWorks.

Marker.WP: Quickly inserts a marker anywhere in your text. AppleWorks' Find Command will find the marker.

Marker.ADB: Lets you permanently mark records in large data bases without changing the data base itself. Makes it easy to locate and move selected records within the data base file.

Marker.ASP: Lets you mark cells in large spreadsheets so you can easily jump to those cells.

Set.Display: Displays your current standard AppleWorks settings on a single screen. Lets you check if one of your macros or another user changed your settings. This is a "must" for teachers and others who share their computers.

Tools.DB: A menu-driven set of eight valuable data base utilities. **Wipe Category** removes the data from a single category in every record. **Wipe File** removes all the data from a data base, leaving only the empty categories and reports. **Edit Category** lets you jump from record to record while staying in the same category. **Fill Category** lets you replace the information in a category in any record or in any set of records you select. **Save Layout** saves the current layout as a report. **Create New Layout** converts your current multiple record layout back to AppleWorks' default settings. **Save As Text File** converts the data base to a text file. **Consecutive Number** inserts consecutive numbers in a category for any set of records that you specify.

Date.Calc: Determines the day of the week for any date you specify. Also calculates the number of days between any two dates you specify.

File.Load: Makes it easy to load a file from any subdirectory into AppleWorks.

Link.SP: Links a spreadsheet to a word processor file so you can write extended notes for the spreadsheet.

Macro.Calc: A menu-driven set of macros that find a common denominator for two numbers, check if a number is a prime number, reduce fractions, convert decimal numbers to fractions, and convert fractions to decimal numbers.

Value.Find: Finds the highest or lowest number in any column or row in your spreadsheet.

Public Domain Update...

WordWrap 4.3: Adds word wrap capability to the AppleWorks data base.

Journal.WP: Makes it easy to use AppleWorks to create and use a daily journal of your activities.

Barrows' Utilities – Disk 16 contains both TimeOut and task file versions of each utility, word processor files with annotated copies of the macros, and documentation in AppleWorks word processor files on the disk. The disk adds important functionality to AppleWorks 4.x and AppleWorks 5. And macro authors will enjoy studying Mr. Barrows' macros to see how he designs and programs these utilities.

Our thanks to Roy Barrows for his continued contributions to the AppleWorks community.

Trees.02

One of the joys of being a **NAUG** member is the opportunity to learn about the different interests and lifestyles of our colleagues. For example, we recently received a letter from long-time **NAUG** member David Francis of Valley View, South Australia. Mr. Francis reports that **NAUG**'s original Trees disk "...has much value as an educational tool...but to a tree lover and hobbyist like me (I have over 50 species of trees in my tiny garden), I'm afraid it just didn't stand up as a Reference work." I wonder what our **NAUG** members who live in large cities think about a colleague who has 50 species of trees in his "tiny garden"!

NAUG members with an interest in horticulture will appreciate Mr. Francis' new Trees.02 disk, which contains an AppleWorks data base with information about 87 different species of trees. We urge members to add trees to this file and to submit their entries to the **NAUG** library; we will add their entries to our master disk.

How to Get Disks

Unless specified otherwise, all **NAUG** disks come in both 5.25-inch ($4) and 3.5-inch ($6) formats. Add $2 s/h *per order* to all prices. All disks are covered by **NAUG**'s "satisfaction guaranteed or your money back" policy. **NAUG** accepts Visa and MasterCard. Order from **NAUG** Public Domain Library, Box 87453, Canton, Michigan 48187; (313) 454-1115; Fax: (313) 454-1965.

Special Offers

NAUG Discount on New Fax Software

Until now, the lack of fax software kept Apple II owners from using the fax capabilities built into their fax modems. But Vitesse's release of FAXination now brings fax capability to your Apple IIGS computer.

FAXination lets you print graphics and text to an external fax modem from any standard GS/OS application. That lets you send documents created with AppleWorks GS, Platinum Paint, and any other GS/OS application to fax machines throughout the world. Vitesse reports that FAXination supports Pointless TrueType fonts, offers convenient phone book capabilities, and maintains a log of faxes sent and received.

FAXination lists for $79.95. Until June 1, 1995 **NAUG** members can buy the program directly from Vitesse for $49.95 plus $5 s/h. (International orders by credit card only; international shipping additional.) Vitesse accepts Visa, MasterCard, and Discover and offers **NAUG** members a 30-day satisfaction guarantee on FAXination.

FAXination requires an Apple IIGS computer running System 5.0.4 or later and equipped with at least two megabytes of RAM, a hard drive, and an external fax modem. System 6.0 or later and four megabytes of RAM are recommended.

At press time, Vitesse is shipping an early release of FAXination. According to the company, version 1.0 has imperfect communications with some modems, does not support batch faxing to a single number, and prints received documents at screen resolution instead of at their original density. Vitesse plans to fix these problems and release free upgrades through the company's bulletin board service.

NAUG will publish a review of FAXination in a future issue of the ***AppleWorks Forum***.

[Vitesse, Inc., Box 929, La Puente, California 91747; (800) 777-7344; Fax: (818) 813-1273.]

NAUG Membership

Name ______________________________

Member N°, if renewing ______________________________

Address ______________________________

____

City ______________________________ State ______________

Zip ______________ Country ______________________________

Home Phone ______________________________

Work Phone ______________________________

The *AppleWorks Forum* — 10 monthly issues, shipped as follows:

Circle One:	One Year	Two[1] Years
2nd Class postage - United States	$30	$60
2nd Class postage - Canada and Mexico	$40	$80
1st Class postage - United States	$46	$92
1st Class airmail - Canada and Mexico	$52	$104
Surface mail outside North America	$44	$88
Airmail outside North America	$67	$134
NAUG on Disk [2]	$90	$180

Total $______________

□ Check □ MC/Visa □ PO # [3] ______________

Credit Card Account # ______________________________

Expiration Date ______________________________

Signature ______________________________

[1] *Avoids future price increases.*
[2] *U.S. Price. International orders by credit card only.*
[3] *Payment must accompany all purchase orders.*

NAUG shares members' addresses with other users groups and selected vendors. If you do not want to receive mail from these agencies, check here: □

AppleWorks is a trademark of Apple Computer, under license to Quality Computers.

Electronic Index Update

Enter the default values for these categories: Volume #: 10 • Issue #:3 • Date: March 1995.

SuperFonts Projects • 1 • How to Preview Your Picture Fonts • Thompson, Bill; Fowler, Jack • fonts; SuperFonts; Graphics; picture fonts

SuperFonts Projects • 6 • Alternate Character Sets • Thompson, Bill; Fowler, Jack • fonts; SuperFonts; graphics; alternate character sets; picture fonts

My Favorite Macro • 7 • How to Convert Spreadsheet Formulas to Values • Johnson, Keith • spreadsheet; formulas; macros

My Favorite Macro • 9 • Another Conversion Macro • Boucher, Robert • spreadsheet; formulas; macros

Special Offers • 10 • Special NAUG Discount on Shareware Solutions II • N/A • Shareware Solutions; newsletters; special offers; Kohn, Joe

AppleWorks 5 Primer • 11 • How to Get Started with AppleWorks 5 • Crutcher, Dan • AppleWorks 5; printers; spelling checkers; macros; TimeOut; UltraMacros; InitManager; inits; mouse; AppleTalk; buffers; custom printers; Ultra – To the Max; task files; printer drivers; crashes

AppleWorks 5 Primer • 15 • Helpful Task File Macros • Crutcher, Dan • AppleWorks 5; macros; task files

AppleWorks 5 Primer • 16 • If You Don't Have the Source File • Crutcher, Dan • AppleWorks 5; macros; task files

AppleWorks News • 17 • ICON Stops Publications • N/A • ICON; TimeOut Central; A2 Central

Special Offers • 17 • Special Discounts from NAUG • N/A • back issues; 1040Works; taxes; 1040Works Tax Planner

Letters to NAUG • 18 • Why Isn't the BBS a Toll-Free Call? • Slaugenhoup, Jeanie • BBS; Electronic Forum

Letters to NAUG • 18 • How to Convert Formulas to Values • Beville, Steve • macros; formulas; spreadsheet

Letters to NAUG • 18 • Printing NLQ Output from AW 5 • Morgan, John • ImageWriter; AppleWorks 5; bugs; printers; printer drivers; interface cards

Letters to NAUG • 19 • Easier Patches for AppleWorks 4 • Serreau, Chris • patches; AppleWorks 4; Apple IIe; Apple IIc; disk management; spelling checkers; crashes; BASIC

Letters to NAUG • 19 • AppleWorks 5 Inits and the Apple IIe • Morgan, John • AppleWorks 5; Apple IIe; inits; macros; Goodies

Letters to NAUG • 20 • How to Update Multiple Records • Woodell, Thomas • data base; Fill Category; Barrows Utilities; Replace Command; macros

Letters to NAUG • 20 • QRAM Test Disk Limits • Katz, Howard • Apple IIe; Apple IIc; QRAM; RAM disks; RamWorks

General Interest • 21 • How to Download NAUG's Files from America Online • Connelly, Joe • America Online; downloading files; Apple File Exchange; Shrink-It; PC Exchange; Macintosh; file transfer

Public Domain Update • 22 • New Disks in the NAUG Public Domain Library • N/A • Trees Disk; Barrows Utilities

Special Offers • 23 • NAUG Discount on FAXination • N/A • FAXination; Vitesse; fax; Apple IIgs

New Keywords: Shareware Solutions; Fill Category; QRAM; downloading files; picture fonts; alternate character sets; Ultra – To the Max; task files; Goodies; Replace Command; PC Exchange; FAXination; Kohn, Joe

Apple-Works Forum

NAUG
National AppleWorks Users Group
Box 87453, Canton, Michigan 48187
(313) 454-1115 Fax: (313) 454-1965
BBS: (615) 359-8238

TIME VALUE MATERIAL